

10-02-00

A

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

ATTY DOCKET NO.: 9219-4

DATE: September 29, 2000

# UTILITY PATENT APPLICATION TRANSMITTAL LETTER AND FEE TRANSMITTAL FORM (37 CFR 1.53(b))

BOX PATENT APPLICATION

Assistant Commissioner for Patents

Washington, DC 20231

Sir:

Transmitted herewith for filing under 37 CFR 1.53(b) is:

- ☒ a patent application  
☐ a Continuation ☐ a Divisional ☐ a Continuation-in-Part (CIP)  
 of prior application no.: ; filed .  
☐ A Small Entity Statement(s) was filed in the prior application; Status still proper and desired.

Inventor(s) or Application Identifier:

Andrew Edward Nunns

Naperville, Illinois 60540-7322

Entitled: **SYSTEMS AND METHODS THAT AUTHORIZE TARGET DEVICES UTILIZING PROPRIETARY SOFTWARE AND/OR HARDWARE**

Enclosed are:

1. ☒ Application Transmittal Letter and Fee Transmittal Form (*A duplicate is enclosed for fee processing*)
2. ☒ 26 pages of Specification (including 50 claims)
3. ☒ 4 sheets of Informal Drawings (35 USC 113)
4. ☒ Oath or Declaration
  - a. ☒ newly executed (*original or copy*)
  - b. ☐ copy from prior application (37 CFR 1.63(d) (*for continuation/divisional*) [Note Box 5 Below]
  - c. ☐ DELETION OF INVENTOR(S) (*Signed statement deleting inventor(s) named in the prior application*)
5. ☐ Incorporation By Reference (*useable if box 4b is checked*)  
 The entire disclosure of the prior application, from which a copy of the oath or declaration is supplied under Box 4b, is considered as being part of the disclosure of the accompanying application and is hereby incorporated by reference therein.
6. ☐ Microfiche Computer Program (*Appendix*)
7. ☐ Assignment papers (*cover sheet(s) and document(s)*)
8. ☐ Small Entity Statement(s)
9. ☒ Information Disclosure Statement, PTO-1449, and 3 references cited
10. ☐ Preliminary Amendment (*Please enter all claim amendments prior to calculating the filing fee.*)
11. ☐ English Translation Document
12. ☐ Certified Copy of

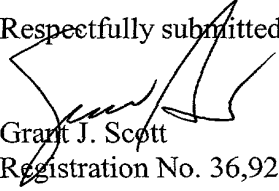
13. ☐ Sequence Listing/ Sequence Listing Diskette  
 a. ☐ computer readable copy  
 b. ☐ paper copy  
 c. ☐ statement in support  
 14. ☐ An Associate Power of Attorney  
 15. ☒ Return Receipt Postcard (MPEP 503) (Should be specifically itemized)  
 16. ☐ Other:

The fee has been calculated as shown below:

	Column 1 No. Filed	Column 2 No. Extra	Small Entity Rate Fee	Large Entity Rate Fee
BASIC FEE			\$345.00	\$690.00
TOTAL CLAIMS	50 - 20 =	30	x 09 = \$	x 18 = \$540.00
INDEP CLAIMS	7 - 3 =	4	x 39 = \$	x 78 = \$312.00
<input type="checkbox"/> MULTIPLE Dependent Claims Presented			+ 130 = \$	+ 260 = \$
If the difference in Col. 1 is less than zero, Enter "0" in Col. 2			Total \$	Total \$1,542.00

- ☒ A check in the amount of \$1,542.00 to cover the filing fee is enclosed.
- ☐ A check in the amount of \$ is enclosed to cover the filing fee, PLUS the Assignment Recordation fee (\$40.00).
- ☒ The Commissioner is hereby authorized to charge payment of the following fees associated with this communication or credit any overpayment to Deposit Account No. 50-0220.
- ☒ Any additional filing fees required under 37 CFR 1.16.
- ☒ Any patent application processing fees under 37 CFR 1.17.

Respectfully submitted,

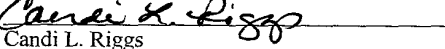
  
 Grant J. Scott  
 Registration No. 36,925

Correspondence Address:  
 USPTO Customer Number: **20792**  
 Myers Bigel Sibley & Sajovec  
 Post Office Box 37428  
 Raleigh, North Carolina 27627  
 Telephone (919) 854-1400  
 Facsimile (919) 854-1401

#### CERTIFICATE OF EXPRESS MAILING

Express Mail Label No.: EL481798323US  
 Date of Deposit: September 29, 2000

I hereby certify that this correspondence is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to Box Patent Application, Commissioner For Patents, Washington, DC 20231.

  
 Candi L. Riggs  
 Date of Signature: September 29, 2000

**SYSTEMS AND METHODS THAT AUTHORIZE TARGET DEVICES  
UTILIZING PROPRIETARY SOFTWARE AND/OR HARDWARE**

Field of the Invention

The present invention relates to integrated systems and methods of operating same, and more particularly, to integrated systems requiring authorization to utilize proprietary software and/or hardware.

Background of the Invention

5 Programmable logic devices (PLDs), such as field programmable gate arrays (FPGAs), typically utilize binary configuration data when performing desired operations in a target application. Frequently, an end user of a programmable logic device would acquire proprietary configuration data ("software") from a software developer by  
10 executing a software license agreement. Pursuant to this agreement, an end user would typically receive either design source code or more often the binary configuration data compiled from the source code. As illustrated by the conventional operations 10 of FIG. 1, proprietary design source code of a developer, Block 12, is typically provided to an end user by  
15 compiling or processing the source code into binary configuration data, Blocks 14, 16. An end user of a PLD would typically then download the configuration data into a programmable read only memory (PROM) mounted on a printed circuit board, Block 18. The configuration data in this memory could then be accessed and loaded into a PLD during start-up  
20 operations, Block 20, so that the PLD becomes configured by the configuration data.

006260" 84492960

Unfortunately, these operations may put the software developer at risk that unauthorized copies of the binary configuration data might be used in additional target applications for which the software developer does not receive compensation. To address this possibility, software developers may develop complex software license agreements to limit unauthorized copying. Such agreements may be difficult to negotiate and may require large up-front royalty fees. Moreover, such terms may preclude users from entering such licenses when only relatively few applications for the proprietary software are anticipated. Thus, notwithstanding conventional licensing techniques for incorporating proprietary software into programmable logic devices, there continues to be a need for improved techniques that do not suffer from the aforementioned limitations associated with conventional software licensing.

#### Summary of the Invention

Preferred integrated systems include devices that authorize programmable logic devices to operate under at least partial control of proprietary software. Each of these "authorization" devices preferably provides continuous or at least periodic authorization to a respective programmable logic device while it is operating in a desired application. This continuous or periodic authorization is preferably provided only so long as the version of software being used by the programmable logic device matches the version of software the authorization device was designed to evaluate and approve.

According to a first preferred embodiment of the present invention, an integrated system comprises an authorization device that generates an encrypted data stream and a programmable logic device (PLD) that also generates an encrypted data stream while simultaneously operating under at least partial control of program code during a first time interval. This program code or software may take the form of data that configures circuitry within the programmable logic device.

0066260" 84292950

Authorization detection circuitry is also preferably provided within the programmable logic device. This circuitry compares the encrypted data streams at least periodically during the first time interval. This circuitry may also disable operation of the programmable logic device if the encrypted data streams indicate that the programmable logic device is not authorized to use the program code. Disabling operation of the programmable logic device may constitute a complete shut down of the programmable logic device or the performance of the programmable logic device may be degraded or impaired sufficiently to render it unacceptable in the desired application.

In particular, the encrypted data streams are evaluated at least periodically during the first time interval to determine whether a "match" is present between the authorization device and the proprietary software used to configure the programmable logic device. A direct ongoing comparison can be made between the encrypted data streams to determine whether there is a sufficiently close identity therebetween while the programmable logic device is operating in a target application. If a sufficiently close identity is present, a "good" flag may be generated within the programmable logic device to enable proper operation for at least some limited time period. An exact identity between the encrypted data streams is preferably not required by the authorization detection circuitry in order to maintain the status of the good flag. However, if a sufficiently close identity is not present between the encrypted data streams over a threshold period of time, then a "fail" flag may be generated. The generation of this fail flag preferably causes the programmable logic device to enter a disabled state. In this disabled state, the programmable logic device may cease to operate or may operate at a degraded or impaired performance level caused by the intentional internal generation of operating errors (e.g., "random" operating errors) by circuitry within the programmable logic device. Other degraded performance states that make the device unfit for the target application may also be possible.

00676748-092900

The above-described authorization scheme may also be applied to other forms of programmable logic devices. Some of these programmable logics devices are frequently referred to by the acronyms PLDs, PLAs, PALs, FPLAs, EPLDs, EEPLDs, LCAs, and FPGAs. In addition, the preferred authorization scheme may be applied to application-specific integrated circuits (ASICs) that perform operations which are exclusively or at least partially hardware based. For example, an ASIC may be designed to perform a plurality of functions and operations useful for a variety of applications. Customers purchasing such ASICs may be able to upgrade or expand the functions and operations performed by the ASIC by purchasing one or more authorization devices at the time the ASIC is purchased or thereafter.

Additional embodiments of the present invention include preferred methods of operating programmable logic devices. These methods preferably include the steps of generating the encrypted data streams during a first time interval while simultaneously operating the programmable logic device under at least partial control of program code that may constitute configuration data. These encrypted data streams are preferably evaluated periodically during the first time interval. The operation of the programmable logic device is then disabled during a second time interval if a comparison of the data streams indicate that the programmable logic device is unauthorized to use the program code.

Brief Description of the Drawings

FIG. 1 is a flow diagram of conventional operations used to load configuration data into programmable logic devices.

FIG. 2 is a flow diagram of exemplary operations used to load configuration data into programmable logic devices.

FIG. 3A is a block diagram of a first integrated system comprising a programmable logic device (PLD) and an authorization device according to a first embodiment of the present invention.

FIG. 3B is a block diagram of a second integrated system comprising an application-specific integrated circuit (ASIC) and an authorization device according to a second embodiment of the present invention.

5                   FIG. 3C is a block diagram of a third integrated system comprising PLDs, ASICs and authorization devices according to a third embodiment of the present invention.

FIG. 4A is a block electrical schematic of preferred deadman circuitry within a PLD or ASIC according to the present invention.

10                   FIG. 4B is a block electrical schematic of a preferred authorization device according to the present invention.

FIG. 4C is a block diagram illustrating operations performed by the stream encryptor of FIG. 4B.

15                   FIG. 5 is a flow diagram of operations that illustrate preferred methods of operating programmable logic devices according to the present invention.

#### Description of Preferred Embodiments

20                   The present invention will now be described more fully hereinafter with reference to the accompanying drawings, in which preferred embodiments of the invention are shown. This invention may, however, be embodied in different forms and should not be construed as limited to the embodiments set forth herein which are provided as preferred examples. Rather, these embodiments are provided so that this disclosure will be thorough and complete, and will fully convey the scope of the invention to those skilled in the art. Like numbers refer to like elements throughout.

25                   Referring now to FIG. 2, exemplary operations **30** for loading a programmable logic device with program code will be described. In particular, FIG. 2 illustrates an operation to generate proprietary design and "deadman" source code, Block **32**. As described more fully  
30                   hereinbelow, this "deadman" source code may augment the design source

006260" 84292950

code so that a programmable logic device performing target operations in accordance with the design code can also be monitored and approved by an authorization device. As illustrated by Blocks **34** and **36**, conventional operations may then be performed to process the source code (including

5 "deadman" source code) into binary configuration data. The binary configuration data may then be provided by the source code developer to a PLD manufacturer or end user. According to the present invention, this conveyance can be made without the need to secure a software license agreement between the conveying party and the receiving party. Instead,

10 preferred authorization devices may be sold to authorize each copy of proprietary software being used in a target application. As illustrated by Block **38**, the binary configuration data may be loaded into a programmable read-only memory (PROM) or another memory storage device that is mounted on a printed circuit board (PCB) along with a

15 respective PLD. The contents of the PROM may then be accessed and downloaded by the PLD using conventional techniques, Block **40**. The downloaded configuration data may operate to configure devices within the PLD that perform operations designed for the target application and configure devices that operate as deadman circuitry. Alternatively, the

20 PLD may acquire the binary configuration data by accessing a network or other device or system external to the PCB.

Referring now to FIGS. 3A-3C, a plurality of integrated systems according to embodiments of the present invention will be generally described. In particular, FIG. 3A illustrates an integrated system

25 **50** comprising a PROM **52** and a preferred PLD **54** having deadman circuitry **54a** therein that becomes programmed by receiving binary configuration data from the PROM **52**. The system also includes an authorization device **56** that provides at least periodic authorization to the PLD **54** to operate under control of the loaded binary configuration data. In

30 FIG. 3B, another preferred integrated system **50'** is illustrated that comprises an application-specific integrated circuit (ASIC) **58** having



deadman circuitry **58a** therein and an authorization device **56** that is electrically coupled to the ASIC **58**. The deadman circuitry **58a** used in ASIC applications typically performs similar operations to the devices within a PLD that are configured as deadman circuitry by the configuration data. In FIG. 3C, an exemplary integrated system **50"** is illustrated that comprises a PLD **54** and an ASIC **58** on an printed circuit board. A plurality of authorization devices **56a-56c** are also provided. As illustrated, these devices may operate in response to a central controller **66**. Additional devices including a PROM **52**, a PLD **60** not requiring authorization, a memory array **62** and I/O circuitry **64** may also be provided. Thus, an integrated system may have one or more PLDs or ASICs and respective authorization devices that operate in conjunction with conventional hardware.

Referring now to FIGS. 4A-4B, a more detailed description of the operation of the integrated systems of FIGS. 3A-3C will be provided. These systems include devices that authorize operation of programmable and other logic devices that operate under at least partial control of proprietary software and/or hardware. With respect to the programmable logic devices **54** illustrated by FIG. 3A, the authorization device **56** preferably provides continuous authorization to a respective programmable logic device **54** while it is operating in a desired target application. This continuous authorization is preferably provided only so long as the binary configuration data ("software") being used by the programmable logic device **54** matches the version of software the authorization device **56** was designed to evaluate and approve.

According to the preferred deadman circuitry **54a** of FIG. 4A, a first data stream P may be generated by a weak random data generator **70**. As illustrated, the weak random data generator **70** may provide a weak random data stream in response to a clock signal and a noise signal and may be of conventional design. An outgoing data framer **72** and an open-drain driver **78** may also be provided so that the first data stream P can be

passed to an input/output pad (I/O). This first data stream P may be provided in-sync with timing signals generated by a central timing circuit **74**. Other formats for the first data stream P may also be used.

Referring now to the preferred authorization device **56** of FIG. 4B, a second encrypted data stream R is generated and provided by a respective stream encryptor **100** to the input/output pad (I/O). This second encrypted data stream R is preferably provided in response to the first data stream P. The illustrated I/O pads associated with the authorization device **56** of FIG. 4B and the deadman circuitry **54a** of FIG. 4A may be connected together by a single-wire bus. A two-wire bus can also be used and such configuration may eliminate the need to provide open-drain drivers. Alternatively, a three-wire bus can be used with an additional clock and such implementation may eliminate the need for timing/framing pulses. If a single-wire bus is used to provide an electrical connection, then the second encrypted data stream R may be time-division multiplexed with the first data stream P on the single-wire bus using a half-duplex format. Operations for multiplexing data in a single-wire bus are generally known to those skilled in the art and need not be described further herein. The illustrated authorization device **56** may comprise an input buffer **90** and an incoming data sampler **92** that operates in response to a central timing circuit **94**. Using these conventional devices, the first data stream P can be retrieved from the I/O pad and provided to an input of the stream encryptor **100**. As described more fully hereinbelow, the stream encryptor **100** of FIG. 4B generates the second encrypted data stream R (with error) using an encryption operation. This encrypted data stream R is then framed and provided to an I/O pad using conventional devices such as the illustrated outgoing data framer **96** and open drain driver **98**. To increase security and to inhibit the likelihood that the construction of the authorization device **56** can be readily reverse-engineered, circuitry within the illustrated stream encryptor **100** may be designed to randomly insert errors into the second encrypted data stream R. The presence of a limited number of intentional

errors will typically increase the difficulty in determining the encryption operation by evaluating the data on the single-wire bus using conventional reverse engineering techniques. If protection against reverse-engineering is not required, then the data streams need not be encrypted.

5 Referring again to the deadman circuitry **54a** of FIG. 4A, an input buffer **80** receives the time-division multiplexed stream (e.g., half-duplex stream) and passes the received stream to an incoming data sampler **76**. In response to central timing, this data sampler **76** extracts the second encrypted data stream R from the multiplexed stream and provides it to an authorization detection circuit (ADC) **84**. As illustrated, the ADC **84** may comprise an exclusive OR (XOR) gate, an error history evaluation circuit and an error timer. The deadman circuitry **54a** may also comprise a stream encryptor **82** that generates and provides a third encrypted data stream R' to an input of the XOR gate within the ADC **84**, in response to the first data stream P and the second encrypted data stream R (with error). To reduce the complexity of the ADC **84**, the stream encryptor **82** within the deadman circuitry **54a** preferably performs operations similar to the stream encryptor **100** within the authorization device **56**, but typically need not perform error insertion operations. This third encrypted data stream R' is preferably generated within the deadman circuitry **54a** (under control of the configuration data compiled from the deadman source code) while other portions of the PLD **54** simultaneously operate under at least partial control of the configuration data compiled from the design source code.

25 According to a preferred aspect of the ADC **84**, a logic 1 error signal is generated at an output of the XOR gate every time a mismatch between the second and third encrypted data streams is detected. A running history of these errors signals is then maintained by the error history circuit. If an insufficient number of errors are detected within a predetermined threshold time period, for example, then a "good" flag may be generated by the ADC **84** to enable proper operation of the PLD **54** for

at least some limited time period. Because of the presence of intentional errors in the second encrypted data stream R, an exact identity between the encrypted data streams is preferably not required by the ADC **84** in order to maintain the status of the good flag. However, if a sufficiently  
5 close identity is not present between the encrypted data streams over a threshold period of time, then a "fail" flag may be generated. The generation of this fail flag preferably causes the PLD **54** to enter a disabled state. In this disabled state, the PLD **54** may cease to operate or may operate at a degraded performance level caused by the intentional internal  
10 generation of operating errors (e.g., "random" operating errors) by circuitry within the PLD **54**. Other degraded performance states may also be possible.

Referring now to FIGS. 4A-4C, encryption operations performed by the deadman circuitry **54a** and the authorization device **56**  
15 will be more fully described. As described above with respect to the weak random data generator **70** in FIG. 4A, a first data stream P may be generated by mixing noise and clock signals. This mixing operation may be performed using conventional techniques using an "unpredictable" circuit that sequentially generates a weak pseudo-random stream of bit  
20 data as  $\{P_1, P_2, P_3, \dots, P_n\}$ , where "n" is an integer. This first data stream P is then provided to a first stream encryptor **82** within the configured deadman circuitry **54a** and also to the authorization device **56**. The authorization device **56** is preferably configured to respond to the first data stream P by generating a second encrypted data stream R of bit data as  
25  $\{R_1, R_2, R_3, \dots, R_n\}$ . This second encrypted data stream R is then fed back, potentially with a limited number of intentional and randomly inserted errors therein, to the deadman circuitry **54a** within the PLD **54**. The second encrypted data stream R may be time division multiplexed (e.g., interleaved) with the first data stream P. Thus, the bit data on the single-  
30 wire bus connecting the authorization device **56** to the PLD **54** may look like:  $\{P_1, R_1, P_2, R_2, P_3, R_3, \dots, P_n, R_n\}$ .

The second encrypted data stream R is preferably generated by performing an encryption operation that evaluates the first data stream P and a plurality of previously generated bits in the second encrypted data stream R. As illustrated by the flow diagram of operations shown in FIG. 4C, the second stream encryptor **100** within the authorization device **56** may use conventional permuting operations to sequentially determine a plurality of permuted bits as  $\{H_1, H_2, H_3, \dots, H_n\}$  during a first time interval, with each permuted bit being determined in accordance with the following expression:

$$H_i = f_p (P_i, R_{i-j}, \dots R_{i-j-k})$$

where  $f_p$  is a permuting function, "i" and "j" are positive integers and "k" represents a preferred "depth" to which the first data stream R is evaluated.

The second stream encryptor **100** within the authorization device **56** may also use a conventional encryption key ( $f_{key}$ ) to generate the second encrypted data stream from the permuted bits in accordance with the following expression:

$$R_{i+1} = f_{key} (H_i, H_{i-l}, \dots H_{i-l-m})$$

where "l" and "m" are positive integers. Other conventional permuting operations and encryption keys may also be used and those described herein are provided as exemplary operations for generating an encrypted data stream.

To increase security and to inhibit the likelihood that the construction of the authorization device can be readily reverse-engineered, error insertion circuitry may be incorporated within the second stream encryptor **100** to intentionally insert "random" errors into the second encrypted data stream R. The presence of a limited number of intentional errors will typically increase the difficulty in reverse engineering the encryption operation by evaluating the data on the single-wire bus.

The first stream encryptor **82** within the deadman circuitry **54a** also preferably performs encryption operations to generate a third

006260" 842960

encrypted data stream R' from the first data stream P and the second encrypted data stream R (with errors). In particular, the exemplary permuting and encryption key operations performed by the first stream encryptor **82** are preferably the same as the corresponding operations performed by the second stream encryptor **100** within the authorization device **56**. However, different operations may also be used by the stream encryptors in less preferred embodiments and the associated authorization detection circuitry may be considerably more complex.

The second and third encrypted data streams R and R' are evaluated at least periodically during the first time interval to determine whether a "match" is present between the authorization device **56** and the proprietary "software" loaded into the PLD **54**. This evaluation is preferably performed by the authorization detection circuitry ADC **84** within the PLD **54**. Thus, a direct ongoing comparison can be made between the encrypted data streams to determine whether there is a sufficiently close identity therebetween, while the PLD **54** is running the proprietary software.

Accordingly, as illustrated by FIG. 5, exemplary operations **110** for authorizing operation of a programmable logic device (PLD) may include operations to generate an at least weakly random data stream, Block **112**, and then generate first and second encrypted data streams within the PLD and authorization device, Blocks **114** and **116**. These encrypted data streams **118** are then evaluated, Block **118**. Exemplary evaluation operations may include a bit-by-bit comparison between the first and second encrypted data streams to determine if a sufficiently close match is present. Then, as illustrated by Block **120**, the PLD is disabled if the evaluation operation fails to indicate authorization of the PLD by the authorization device.

The above-described authorization scheme may also be applied to other forms of programmable logic devices (PLDs). Some of these programmable logics devices are frequently referred to by the acronyms PLAs, PALs, FPLAs, EPLDs, EEPLDs, LCAs, and FPGAs. In

09676748-092900

addition, the preferred authorization scheme may be applied to application-specific integrated circuits (ASICs) that perform operations which are exclusively or at least partially hardware based. For example, the ASIC **58** of FIGS. 3B and 3C may be designed to perform a plurality of functions and operations useful for a variety of target applications. Customers purchasing such ASICs may be able to upgrade or expand the functions and operations performed by the ASIC **58** by purchasing one or more authorization devices at the time the ASIC **58** is purchased or thereafter. According to one embodiment applicable to ASICs, each of these additional authorization devices **56b**, **56c** can be connected to respective pins of the ASIC **58** to enable the ASIC **58** to perform additional or replacement functions and operations that correspond to the particular authorization device or combination of authorization devices. Alternatively, multiple authorization devices could be configured to share a common bus line, and in this case the respective ASIC **58** could be designed to have a set of chip selects for a plurality of authorization devices. The protocol could also be extended to deal with multiple authorization devices sharing a common bus line without individual selects.

Thus, an additional embodiment of the present invention may include first and second integrated circuit devices that generate first and second data streams, respectively, while the first integrated circuit device (e.g., ASIC, PLD) performs software and/or hardware controlled operations. This first integrated circuit device preferably has authorization detection circuitry therein that receives and at least periodically evaluates the first and second data streams and disables the software and/or hardware controlled operations when the first and second data streams fail to indicate a sufficient match between the second integrated circuit device (e.g., authorization device) and the software and/or hardware controlled operations performed by the first integrated circuit device. Still further embodiments of the present invention may include "slave" PLDs (or slave ASICs) that monitor the single-wire bus between a master PLD (or master

006260 8429960

ASIC) and a respective authorization device. Each slave device may listen to the data provided on the single-wire bus to determine whether authorization is occurring. This determination may be made by incorporating into each slave device deadman circuitry that is similar to the circuitry within a corresponding master device. A slave device need not have circuitry to enable it to generate the first data stream P on the single-wire bus, however, additional circuitry may be necessary to enable it to operate in-sync with the communications between the master device and the authorization device.

10                   In the drawings and specification, there have been disclosed typical preferred embodiments of the invention and, although specific terms are employed, they are used in a generic and descriptive sense only and not for purposes of limitation, the scope of the invention being set forth in the following claims.



THAT WHICH IS CLAIMED IS:

1. An authorization device, comprising:

an integrated circuit component that in response to a first data stream generates a second encrypted data stream which is at least periodically evaluated during a first time interval to assess whether operation of a programmable logic device during the first time interval is authorized.

5

2. The authorization device of Claim 1, wherein the first data stream and the second encrypted data stream are time division multiplexed on an I/O pin associated with said integrated circuit component.

3. The authorization device of Claim 1, wherein said integrated circuit component utilizes an encryption operation to generate the second encrypted data stream from the first data stream.

4. The authorization device of Claim 3, wherein said integrated circuit component comprises circuitry that intentionally inserts errors into the second encrypted data stream.

5. The authorization device of Claim 3, wherein the encryption operation generates a first permuted bit as a function of a first bit in the first data stream and at least a first encrypted bit in the second encrypted data stream.

6. The authorization device of Claim 5, wherein the encryption operation uses an encryption key to generate a second encrypted bit in the second encrypted data stream from at least the first permuted bit.

7. The authorization device of Claim 6, wherein the encryption operation generates a second permuted bit as a function of a second bit in the first data stream and at least the second encrypted bit in the second encrypted data stream.

8. The authorization device of Claim 7, wherein the encryption operation uses the encryption key to generate a third encrypted bit in the second encrypted data stream from the second permuted bit and the first permuted bit.

9. The authorization device of Claim 4, wherein the encryption operation generates a first permuted bit as a function of a first bit in the first data stream and at least a first encrypted bit in the second encrypted data stream.

10. The authorization device of Claim 9, wherein the encryption operation uses an encryption key to generate a second encrypted bit in the second encrypted data stream from at least the first permuted bit.

11. The authorization device of Claim 2, wherein said integrated circuit component utilizes an encryption operation to generate the second encrypted data stream from the first data stream.

12. The authorization device of Claim 11, wherein the encryption operation generates a first permuted bit as a function of a first bit in the first data stream and at least a first encrypted bit in the second encrypted data stream.

13. The authorization device of Claim 12, wherein the encryption operation uses an encryption key to generate a second encrypted bit in the second encrypted data stream from at least the first permuted bit.

14. The authorization device of Claim 1, wherein the first data stream is an at least weakly random sequence of bits.

15. The authorization device of Claim 4, wherein the first data stream is an at least weakly random sequence of bits.

16. An integrated system, comprising:

an authorization device that generates a first encrypted data stream;

a programmable logic device that generates a second encrypted data stream while simultaneously operating under at least partial control of configuration data during a first time interval; and

authorization detection circuitry that at least periodically compares the first and second encrypted data streams during the first time interval and disables operation of said programmable logic device if the first and second encrypted data streams indicate that said programmable logic device is not authorized to utilize the configuration data.

17. The system of Claim 16, wherein said programmable logic device generates an at least weakly random data stream during the first time interval; and wherein said authorization device generates the first encrypted data stream in response to the at least weakly random data stream.

18. The system of Claim 16, wherein said authorization detection circuitry is internal to said programmable logic device; wherein said programmable logic device utilizes an encryption operation to generate the second encrypted data stream; and wherein each of a plurality of bits in the second encrypted data stream is determined by evaluating at least one bit in the first encrypted data stream.

006260" 84292960

19. The system of Claim 17, wherein said authorization detection circuitry operates as a dead man switch internal to said programmable logic device; wherein said programmable logic device utilizes an encryption operation to generate the second encrypted data stream; and wherein  
5 each of a plurality of bits in the second encrypted data stream is determined by performing the encryption operation on at least one respective bit in the first encrypted data stream and at least one respective bit in the at least weakly random data stream.

20. The system of Claim 19, wherein each of the plurality of bits in the second encrypted data stream is determined at a respective point in the first time interval by performing the encryption operation on at least one bit in the first encrypted data stream generated at an earlier point in the time  
5 interval and at least one bit in the at least weakly random data stream.

21. An integrated system, comprising:  
an authorization device that generates a first encrypted data stream;  
an integrated circuit device that generates a second encrypted data stream and performs first operations during a first time interval; and  
5 authorization detection circuitry that at least periodically compares the first and second encrypted data streams during the first time interval and disables operation of said integrated circuit device if the first and second encrypted data streams indicate that said integrated circuit device is not authorized to perform the first operations.

22. The system of Claim 21, wherein said integrated circuit device generates an at least weakly random data stream during the first time interval; and wherein said authorization device generates the first encrypted data stream in response to the at least weakly random data  
5 stream.

09675748-092900

5 23. The system of Claim 21, wherein said authorization detection circuitry is internal to said integrated circuit device; wherein said integrated circuit device utilizes an encryption operation to generate the second encrypted data stream; and wherein each of a plurality of bits in the second encrypted data stream is determined by evaluating at least one bit in the first encrypted data stream.

5 24. The system of Claim 22, wherein said authorization detection circuitry operates as a dead man switch internal to said integrated circuit device; wherein said integrated circuit device utilizes an encryption operation to generate the second encrypted data stream; and wherein each of a plurality of bits in the second encrypted data stream is determined by performing the encryption operation on at least one respective bit in the first encrypted data stream and at least one respective bit in the at least weakly random data stream.

25. The system of Claim 22, wherein said authorization device and said integrated circuit device are electrically connected together by a bus; and wherein the at least weakly random data stream is time division multiplexed on the bus with the first encrypted data stream.

26. A method of operating a programmable logic device, comprising the steps of:

generating first and second encrypted data streams in first and second devices, respectively, while simultaneously operating the programmable logic device configured to perform a first operation during a first time interval; and

evaluating the first and second encrypted data streams at least periodically during the first time interval and disabling operation of the programmable logic device during a subsequent second time interval if a comparison of the first and second data streams indicate that the programmable logic device is not authorized to perform the first operation.

27. The method of Claim 26, further comprising the step of generating an at least weakly random data stream during the first time interval; and wherein the first and second encrypted data streams are generated from the at least weakly random data stream.

28. The method of Claim 27, wherein the first encrypted data stream is generated internal to the programmable logic device and the second encrypted data stream is generated external to the programmable logic device.

29. The method of Claim 28, wherein the at least weakly random data stream is generated internal to the programmable logic device; wherein the at least weakly random data stream is provided by a single wire bus to a device external to the programmable logic device; and wherein the at least weakly random data stream is time division multiplexed on the bus with the second encrypted data stream.

30. The method of Claim 29, wherein the at least weakly random data stream is generated by mixing clock and noise signals.



36. The device of Claim 35, wherein said first integrated circuit component comprises circuitry that intentionally inserts random errors into the second encrypted data stream.

37. An integrated circuit system, comprising:

a first component that in response to a first data stream generated external to said first component generates a second encrypted data stream; and

5 a second component that at least periodically evaluates the second encrypted data stream to assess whether performance of at least one operation within the second component is authorized during a time interval when the first data stream is being generated.

38. The system of Claim 37, wherein said second component comprises an integrated circuit selected from the group consisting of ASICs and PLDs.

39. The system of Claim 37, wherein said second component generates the first data stream; and wherein said first and second components comprise first and second stream encryptors therein, respectively.

40. The system of Claim 37, wherein said first and second components are electrically connected together by a single wire bus; and wherein the first data stream and the second encrypted data stream are time division multiplexed on the single wire bus.

41. The system of Claim 39, wherein said first and second components are electrically connected together by a single wire bus; and wherein the first data stream and the second encrypted data stream are time division multiplexed on the single wire bus.



42. The system of Claim 41, wherein said first component comprises circuitry that intentionally inserts random errors into the second encrypted data stream.

43. The system of Claim 39, wherein the second encryptor within said second component generates a third encrypted data stream; and wherein said second component comprises circuitry that operates as a deadman switch to disable performance of the at least one operation within said second component if the second and third encrypted data streams fail to indicate that said second component is authorized by said first component to perform the at least one operation.

44. An integrated circuit system, comprising:  
first and second integrated circuit devices that generate first and second data streams, respectively, while said first integrated circuit device performs software and/or hardware controlled operations, said first integrated circuit device having authorization detection circuitry therein that receives and at least periodically evaluates the first and second data streams and disables the software and/or hardware controlled operations when the first and second data streams fail to indicate a sufficient match between said second integrated circuit device and the software and/or hardware controlled operations performed by said first integrated circuit device.

45. The system of Claim 44, wherein said first and second integrated circuit devices generate the first and second data streams in response to an at least weakly random sequence of bits.

46. The system of Claim 45, wherein said first and second integrated circuit devices are electrically coupled together by a single wire bus; and wherein the at least weakly random sequence of bits and the second data stream are time division multiplexed on the single wire bus.

5 47. The system of Claim 46, wherein said first integrated circuit device comprises a first stream encryptor that generates the first data stream as a first encrypted data stream from the at least weakly random sequence of bits; and wherein said second integrated circuit device comprises a second stream encryptor that generates the second data stream as a second encrypted data stream from the at least weakly random sequence of bits.

48. The system of Claim 47, wherein said first integrated circuit device comprises authorization detection circuitry that generates an error history from the first and second encrypted data streams.

49. The system of Claim 48, wherein said second integrated circuit device comprises circuitry that intentionally inserts random errors into the second encrypted data stream.

50. The system of Claim 45, wherein said first integrated circuit device generates the at least weakly random sequence of bits and comprises a first stream encryptor that generates the first data stream as a first encrypted data stream from the at least weakly random sequence of bits and the second data stream; and wherein said second integrated circuit device comprises a second stream encryptor that generates the second data stream as a second encrypted data stream from the at least weakly random sequence of bits.

5

**SYSTEMS AND METHODS THAT AUTHORIZE TARGET DEVICES  
UTILIZING PROPRIETARY SOFTWARE AND/OR HARDWARE**

Abstract of the Disclosure

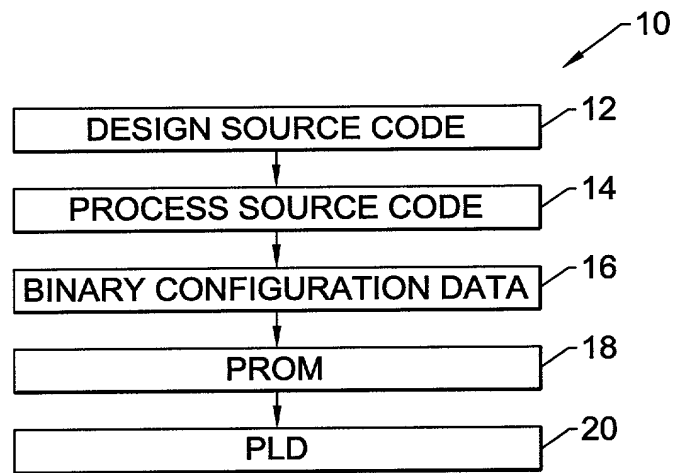
09676748-092900

An integrated system comprises an authorization device that generates a second encrypted data stream in response to a first data stream, and a programmable logic device (PLD) that generates a third encrypted data stream in response to the first data stream, while

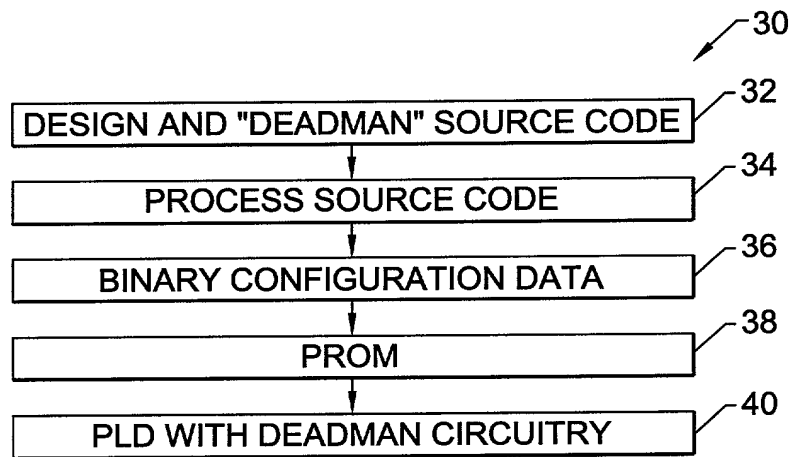
5 simultaneously operating under at least partial control of program code during a first time interval. This third encrypted data stream is preferably generated internal to the programmable logic device. Authorization detection circuitry is also preferably provided that compares the second and third encrypted data streams at least periodically during the first time

10 interval. This circuitry may also disable operation of the programmable logic device if the second and third encrypted data streams indicate that the programmable gate array is not authorized to use the program code. The authorization detection circuitry is preferably provided within the programmable logic device and may utilize at least a portion of the

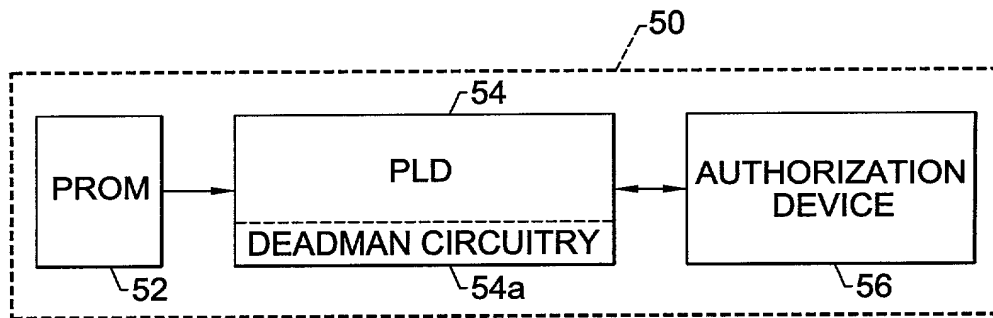
15 proprietary program code (e.g., "deadman" code) to perform its operations. Disabling operation of the programmable logic device may constitute a complete shut down of the programmable logic device or the performance of the programmable logic device may be degraded sufficiently to render it unacceptable in the desired application.



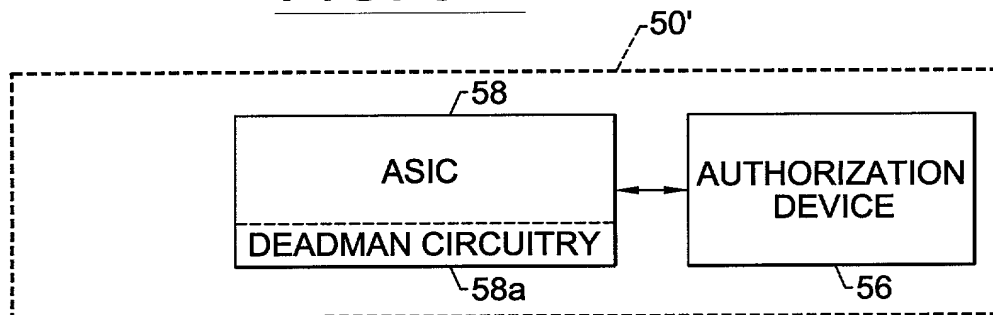
**FIG. 1.**  
(PRIOR ART)



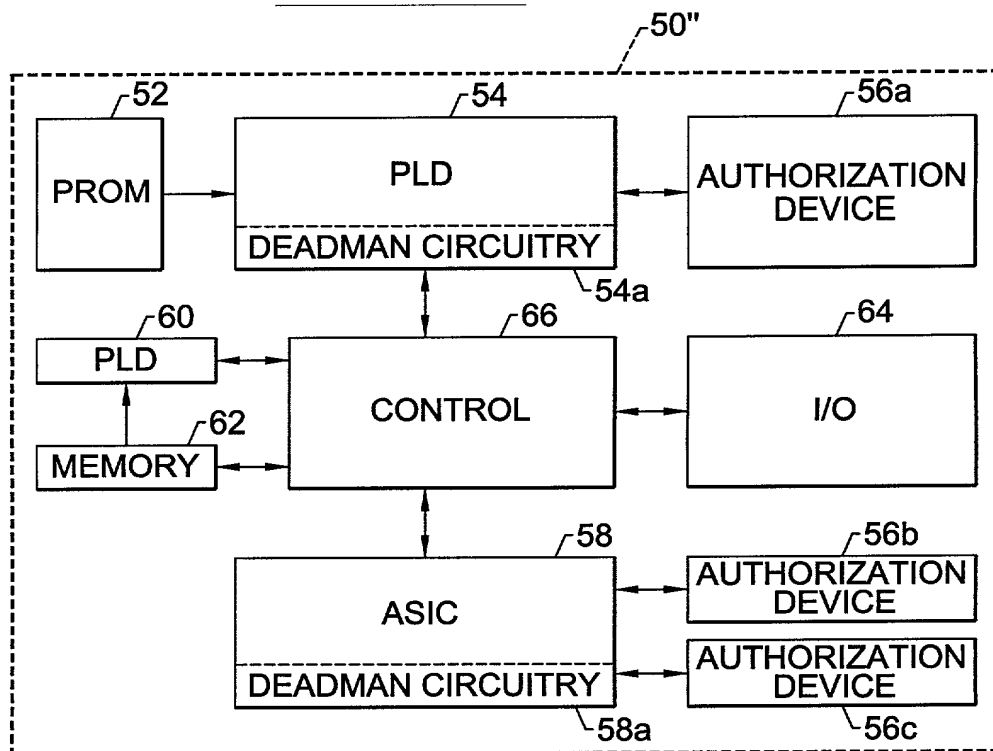
**FIG. 2.**



**FIG. 3A.**



**FIG. 3B.**



**FIG. 3C.**

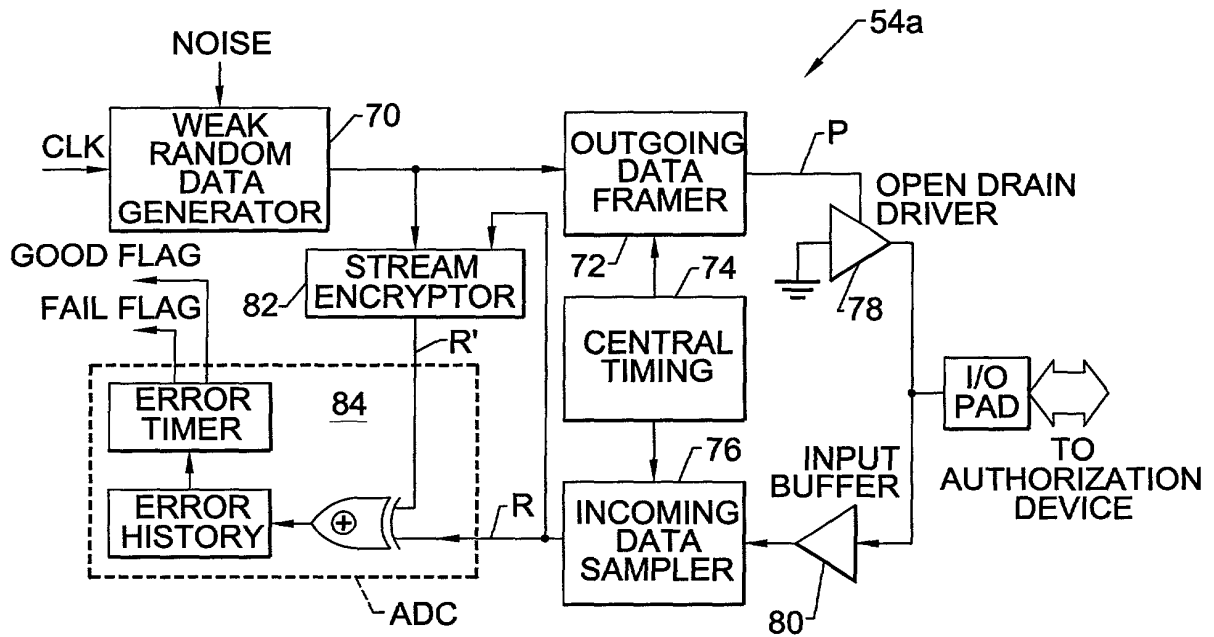


FIG. 4A.

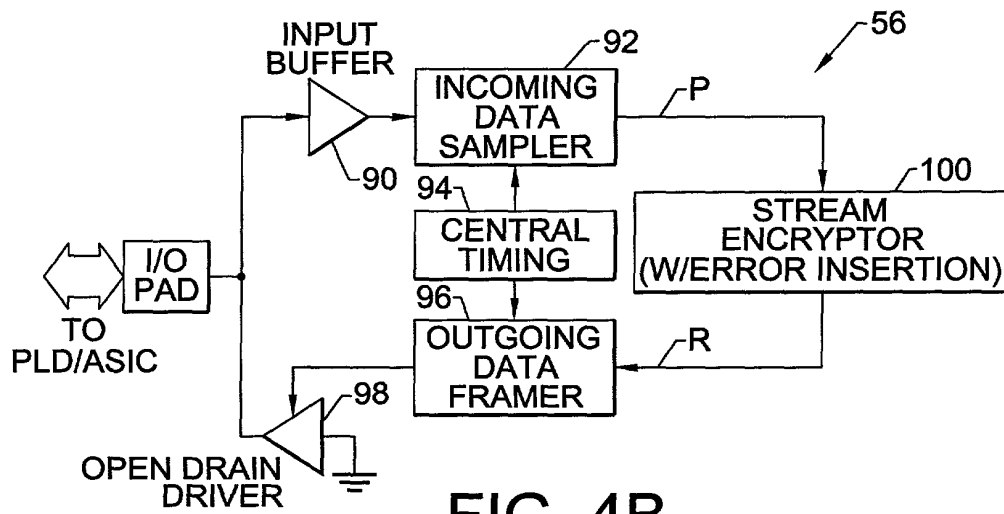


FIG. 4B.

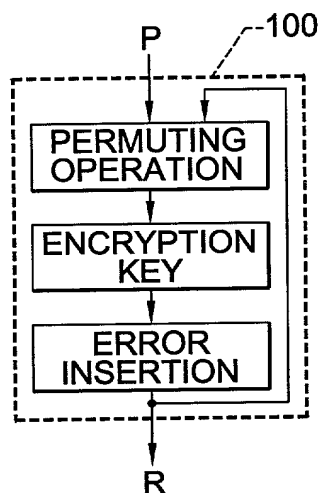


FIG. 4C.

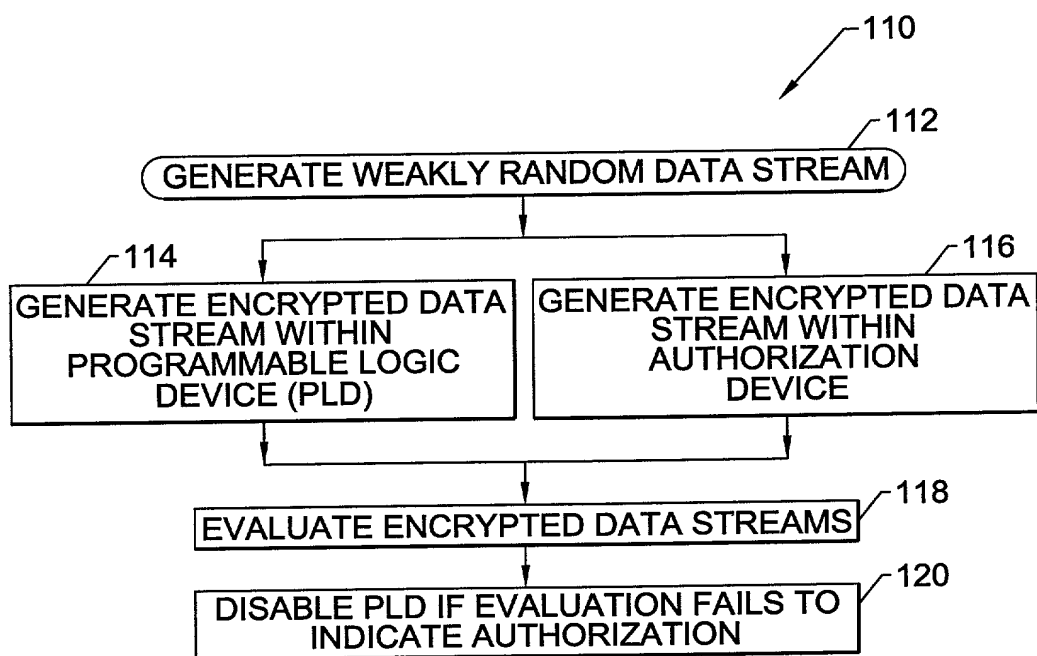


FIG. 5.



**DECLARATION AND POWER OF ATTORNEY FOR PATENT APPLICATION**  
Attorney Docket No. 9219-4

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name.

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled **SYSTEMS AND METHODS THAT AUTHORIZE TARGET DEVICES UTILIZING PROPRIETARY SOFTWARE AND/OR HARDWARE**,

the specification of which

☒ is attached hereto

OR

☐ was filed on \_\_\_\_\_ as United States Application No. or PCT International Application Number \_\_\_\_\_ and was amended on \_\_\_\_\_ (if applicable).

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to patentability as defined in Title 37 Code of Federal Regulations, §1.56.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

**POWER OF ATTORNEY:** As a named inventor, I hereby appoint the following registered attorney(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith.

Daniel N. Yannuzzi  
Registration No. 36,727

James K. Dawson  
Registration No. 41,701

Kelly H. Hale  
Registration No. 36,542

Robert P. Hart  
Registration No. 35,184

Keith Kind  
Registration No. 42,735

Semion Talpalatsky  
Registration No. 35,380

Grant J. Scott  
Registration No. 36,925

D. Scott Moore  
Registration No. 42,011

**Customer Number 20792**

Send correspondence to: Grant J. Scott  
Myers Bigel Sibley & Sajovec  
Post Office Box 37428  
Raleigh, NC 27627

Direct telephone calls to: Grant J. Scott  
(919)854-1400

Facsimile: (919) 854-1401

Full name of sole inventor: **Andrew Edward Nunns**

Inventor's  
Signature: Andrew E. Nunns Date: 27 Feb 2000

Residence: Naperville, Illinois

Citizenship: United States of America

---

Post Office Address: 339 Robin Hill Drive  
Naperville, Illinois 60540-7322

09676748-0929000